XIV Jornadas de Ingeniería Telemática.
JITEL 2019.
Universidad de Zaragoza.

Actas de las XIV Jornadas
de Ingeniería Telemática
(JITEL 2019),
Zaragoza (España),
22-24 de octubre de 2019.

ISBN: X-XXXX-XX-XXX-X
DOI: XXXXXX

A Novel Admission Control Scheme for Network Slicing based on Squatting and Kicking Strategies

Ahmed El-mekkawi, Xavier Hesselbach, and Jose Ramon Piney
Dept. Network Engineering,
Universitat Politecnica de Catalunya
C/ Jordi Girona, 1-3 - Edif.C3 - Campus Nord - 08034 Barcelona - Spain
(ahmed.mohamed.abdelaty.elmekaw, xavier.hesselbach)@upc.edu, jpiney@entel.upc.edu

***Abstract*—New services and applications impose different quality of service (QoS) requirements on network slicing. To meet differentiated service requirements, current Internet service model has to support emerging real-time applications from 5G networks. The admission control mechanisms are expected to be one of the key components of the future integrated service Internet model, for providing multi-level service guarantees with the different classes (slices) of services. Therefore, this paper introduces a new flexible admission control mechanism, based on squatting and kicking techniques (SKM), which can be employed under network slicing scenario. From the results, SKM provides 100% total resource utilization in bandwidth context and 100% acceptance ratio for highest priority class under different input traffic volumes, which cannot be achieved by other existing schemes such as AllocTC-Sharing model due to priority constraints.**

***Index Terms*—SKM, Admission Control, Class of Service, Utilization Optimization**

I. INTRODUCTION

With the emergence of network slicing in 5G, and network virtualization embedding strategies, resource management models are required to provide 100% utilization in a multi-class context under bandwidth constraints [1]. Furthermore, as the demand for different types of services and applications increases, integrating the services into the Internet will have a profound influence on the future extension of Internet networking technologies. Hence, the diversified applications with different QoS requirements are considered to be the most important components of the future IP services under 5G networks [2] [3]. Under the motivation of the rapid growth of real-time service requirements, the current Internet is smoothly shifting from the best-effort network into an integrated services network, little by little. Recently, more and more emerging Internet real-time applications that, require more than best-effort service are increasingly being carried out on the Internet [4] [5]. Moreover, the network operator wants to maximize the revenue by increasing the number of users without compromising the promised Quality of service. This can only be achieved by efficient admission control model that directly controls the number of users admitted into the system. In this regard, Bandwidth Allocation Models (BAMs) that have been proposed in the past to set application requirements and priorities over a range of traffic classes, can serve as models for admission control. BAMs establish the amount of bandwidth per-class and any eventual sharing among them [6]. Moreover, BAMs can handle any type of resources allocation [7]. In the literature, several works deal with the dynamic bandwidth allocation for guaranteeing a given QoS level per class and optimizing the utilization. These contributions are based on the Maximum Allocation Model (MAM) [8], Russian Doll Model (RDM) [9], Generalized RDM (G-RDM) [10], AllocTC-Sharing model (AllocTC) [11] among others. Fig. 1 illustrates examples of MAM, RDM, G-RDM and AllocTC allocation algorithms for three CTs, where the RC (resource constraints) value corresponds to the bandwidth restriction (limit) imposed to one or more CTs. MAM is a strict

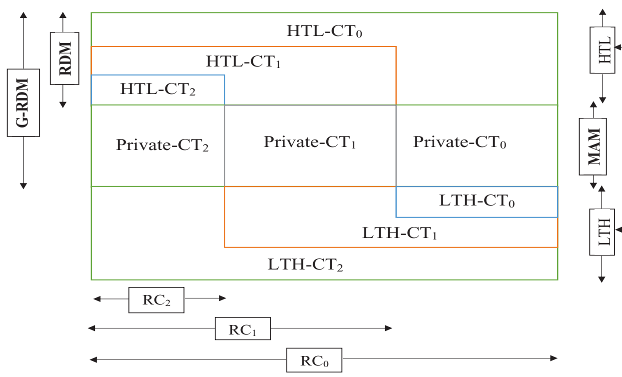


Fig. 1: BAMs and resource allocation strategies [12]

.

allocation model in which another class type (CT) cannot share (private resources) the unused bandwidth of a given CT. On the other hand, RDM is a nested allocation model where non-utilized bandwidth allocated to the higher hierarchical CTs might be used by lower priority CTs temporarily (High to Low loan - HTL loan). Moreover, AllocTC model allows an opportunistic sharing of the bandwidth between the different classes. It is considered as an enhancement of the RDM model because it not only allows an HTL loan but Low to High loan

(LTH loan) as well. G-RDM is a hybrid model in which the "HTL loan" strategy of RDM incorporates the private resource strategy defined by MAM. However, these models do not take into account various Service Level Agreements (SLA) such as latency, packet loss, and jitter to adjust bandwidth, and they can not guarantee higher admission for high priority classes after network congestion. Therefore, the main contribution of this paper proposes to integrate all of these models in a single admission control model, in multi-class networks being able to provide 100% total resource utilization based on squatting and kicking strategies that can work under offline and online scenarios. In offline scenario, all demands are known in advance without lifetime constraint, while in online scenario, demands arrive on a real-time basis with a specific lifetime. SKM, guarantees high admission for QoS of higher priority classes under different input traffic volumes, especially in congested scenarios (i.e. such as video, if it is more important than others in a network, then by using the SKM, a network administrator can prioritize video traffic to ensure that the service remains uninterrupted, while the other traffic may be suspended or even dropped). On the other hand, for the case of uncongested scenarios, the SKM behaves similar to MAM, RDM and AllocTC.

Moreover, SKM is a suitable strategy for emerging technologies that are charactered by diverse QoS requirements and prioritized admission control. The concept of QoS allows certain types of traffic to be prioritized in the network. A case at hand will be network slicing scenario, where the different slices have varying priorities in terms of admission and resource allocation.

The remainder of this paper is organized as follows: In section II, related works are listed. In section III, we present the definition and the description of SKM proposal, including SKM scenarios in offline and online mode. Section IV describes performance evaluation issues. Section V presents the obtained results and discussion. Section VI concludes the paper and presents future work.

II. RELATED WORKS

BAMs are of great value in the context of efficient and customized use of resources management. Moreover, BAMs can work as admission control models. Several works based on BAMs dealt with the dynamic bandwidth allocation for guaranteeing a given QoS level per CT and optimizing the utilization. In [13], the authors propose a method of dynamic and hierarchical allocation of the bandwidth using RDM strategy. This method is based on the classification and the prioritization of services. The algorithm provides the bandwidth required for the demands based on fairness factor and services priority.

The general problem of the algorithms based on RDM is that the resource reservation is carried out from bottom to top; the lower priority traffic shares its resources with higher priority traffic and not the inverse. Several works have been carried out proposing new dynamic bandwidth sharing algorithms by adopting the RDM strategy [14] [15].

To make the reservation from top to bottom and from bottom to top, the AllocTC [11] initiated two-way algorithm of dynamic bandwidth sharing, where unused bandwidth of high priority CTs can be shared with low priority CTs. In [7] the authors studied the behaviour and resource allocation characteristics of the BAMs, then they compared distinct BAMs using different traffic scenarios. The authors proved by simulation that AllocTC is more efficient in terms of optimizing the utilization of the link and that it is better suited for elastic traffic and high bandwidth utilization. The authors in [12] propose a new approach with a combination of (MAM, RDM, G-RDM, and AllocTC) models based on a controller by using different metrics to switch from one model to another one in order to improve the efficiency of the performance for instance link utilization, blocking probability, and packet number. In [16], the authors proposed a new model called (smart AllocTC), which runs on a controller to manage the QoS and routing with QoS constraints. The model applies RDM and AllocTC strategies to classify demands based on their threshold severity (high, medium, and low). Whenever the priority of demand is of the high threshold, the (smart AllocTC) benefits from other categories bandwidth and calculates the fairness index of the categories to allocate resources precisely to all demands taking into account their priorities.

However, all these models cannot give 100% total resource utilization and guarantee higher admission for higher priority classes at same time.

III. SQUATTING AND KICKING MODEL (SKM) PROPOSAL

The need for network slicing and network virtualization for 5G networks requires an admission control model that can support fast and dynamic discovery of the resources that will often be heterogeneous in type, implementation, and independently administered. Thus, the main idea of our proposed admission control model exploits resources partition and reservation, according to different priority classes with the flexibility of using the full amount of resources when other CTs do not demand them. Furthermore, SKM provides a smoother BAM policy transition among existing policy alternatives resulting from MAM, RDM, AllocTC adoption independently in a single solution, to improve the utilization and to guarantee high admission for the higher priority CTs. This strategy is used as an admission control function for highly congested scenarios, with strict constraints for the higher priority CTs. On the other hand, for the case of uncongested scenarios, then the SKM behaves similar to classical BAM techniques.

A. Definitions

Traffic Classes - TC (also CT or class or class of service COS) according to RFC 4127: is a logical group of demands that meet a given resources constraint, such as equal value in a specific header field (e.g. source-destination) [9]. TC populate the so-called multi-class networks.

Squatting: action of occupying resources allocated to other (higher or lower) classes when their holders are not using them.

Kicking: action of expelling a lower priority class from its allocated resources, either partially or totally [17].

B. Assumptions and Notations

The goal of the auto-provisioning, SKM model is to achieve more efficient admission control mechanism for prioritized user demands. The proposed model jointly considers the priorities of both admitted and arriving demands and the current resource utilization in the system. In this work, the contested resources of single link can support up to R, which represents the capacity of the resource of the system; the size of the R can be discrete or continuous. R is partitioned in classes, N is the number of classes defined in the link, and where RCc is the maximum reservable resources in class c, as shown in Fig. 2.

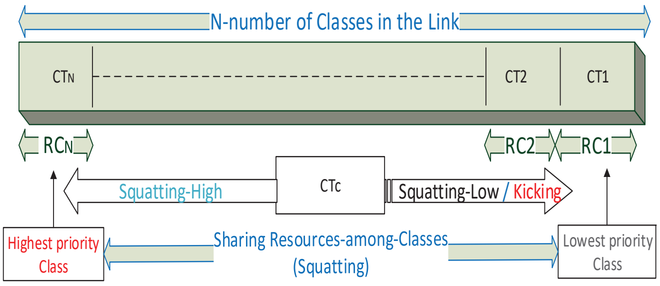


Fig. 2: SKM-Strategy

C. Algorithm Setting

A description of all parameters and decision variables used in our admission control is provided in Table I and Table II respectively.

TABLE I: Parameters of the Model

Abbreviation	Explanation
RC_c	Resource Constraints for class c also equal to maximum reservable resources for class c
CT_c	Class of priority c where $c \in [1, N]$ and CT_N is the highest priority class and CT_1 is the lowest priority class.
R	Maximum allocable resources for all classes together and is equal to link capacity
$d_j(CT_c)$	The amount of resources (size) of demand j belonging to class c where $j \in [1, D]$

TABLE II: variables of the Model

Abbreviation	Explanation
D	Total Number of demands by all classes
D_c	Total Number of demands by class c
S_c	The actually allocated resources to class c
BD	Number of blocked demands by all classes
BD_c	Number of blocked demands by class c
AD	Number of accepted demands by all classes
AD_c	Number of accepted demands by class c
P_{LTH}	The number of preemption of higher priority traffic by lower priority traffic
P_{HTL}	The number of preemption of lower priority traffic by higher priority traffic
SH_i	Squatted resources from higher priority class i
SL_i	Squatted resources from lower priority class i
K_i	Kicked resources from lower priority class i

D. Conceptual model behavior

Different strategies such as Squatting model, MAM, RDM, GRAM, AllocTC and others can be considered, depending on performance and goals provided by each strategy. In our proposed model, the sharing approach already used by MAM, RDM, AllocTC and the SKM allows CTs with higher priority to use available resources allocated to lower priority CTs and vice versa. Unlike other BAMs, in SKM, if a given CT of service requires more resources than those allocated to it, the procedure of the model, for each demand, will be as follows:

- SKM starts working as a normal MAM algorithm (step 1).
- If resources are not enough, SKM check where resources are not used, starting with higher priority classes (Step 2). This is similar to squatting of the higher priority classes (Sq-H) or RDM style.
- Else, if more resources are required, SKM check where resources are not used from lower priority classes (Step 3). This is similar to squatting of the lower priority classes (Sq-L) or loan of lower priority traffic by higher priority traffic (AllocTC style).
- Else, try using Kicking and count the kicked class in the blocking probability for the same class.
- Else, the demand cannot be allocated.

Based on the service policy, the Squatting technique aided by its two priority classes (high and low) to be less aggressive than kicking technique, especially in case of the uncongested scenarios. Therefore squatting technique is generally preferred over kicking, if the class requires extra resource allocation, as shown in Alg 1.

Algorithm 1 Process Assignment algorithm for SKM

procedure PROCESS ASSIGNMENT(Loop D :Demands; Loop Demands)
  for Each Demand $d_l = d_l(CT_i) \in D$ **do**
    if $d_l \leq RC_i$ **then** ▷ Strategy MAM
      Allocate d_l resources from the class i
    else if $\exists j \;\; s.t. \;\; j > i \wedge d_l \leq CT_j$ $\mathit{Available\ resources}$ **then** ▷ Strategy RDM or Squatting-High
      Allocate d_l resources from CT_j ▷ SH_j
    else if $\exists j$ where $j < i \;\; s.t. \;\; d_l \leq CT_j$ $\mathit{Available\ resources}$ **then** ▷ Squatting-Low
      Allocate d_l resources from (CT_j) ▷ SL_j
    else
      found-kick=false
      for j=1 to i-1 **do**
        if ¬(found-kick) and ($\exists d_m(CT_n) \in (CT_j)$, and , n<i) **then**
          kick $d_m(CT_n)$ from (CT_j) ▷ found-kick=true
        end if
      end for
      if ¬(found-kick) **then**
        Reject d_l
      end if
    end if
  end for
end procedure

E. Offline and online scenarios

The proposed algorithm in this paper was designed to work as admission control for offline and online scenarios. In offline scenario, all demands are known in advance, and they do not have lifetime constraint (i.e. allocated without expiry limit). While in online scenario, demands arrive on real-time basis with specific arrival and expiry times. The following paragraphs introduce the overall idea of each scenario as follow:

1) Offline scenario: The goal of SKM performance is to make the best selection of user demands to be admitted considering, user priorities and available resources in the system.

The SKM offline behaviour introduces a new method for deciding the demands that can be admitted. In this scenario, we simplify the procedure of checking demands by arranging them according to their priorities and sizes. Based on that, the high priority classes will be allocated to the high priority demands first, and then low priority classes can be allocated to the remaining demands if there are enough resources.

2) Online scenarios: In the SKM performance of the online scenario, the traffic of the system can be distributed fairly according to the QoS policy. This provides efficient usage of system resources and solves the online allocation problems such as the rerouting of the demands according to the priority along the unit times. In the online mode, the demands are sorted according to size and priority to minimize the number of kicking operation. The difference between the SKM behaviour in offline mode and online mode is that in the offline mode the sorting process performed once before the allocation process. In online mode, the sorting is done before the process of the assignment of the demands in each unit time as in Alg 1.

IV. PERFORMANCE EVALUATION

This section presents technical comparison of SKM against the state of the art algorithms. Also, an evaluation methodology is presented, which includes performance metrics and descriptions of simulation scenarios.

A. Technical behavior and other operational characteristics

Table III shows a set of possible behaviours and operational characteristics adopted to manage system resources for admitting user demands. In other words, to obtain expected use and accept demands depending on available resources and traffic load using SKM and other comparative models.

B. Offline evaluation metrics

The evaluated metrics for permanent demands addressed in this paper is the total acceptance ratio (AR), total utilization (U), acceptance ratio per class (AR_c) and utilization per class (U_c) according to Table IV as below:

TABLE IV: Offline metrics definitions

Abbreviation	Explanation
Acceptance ratio AR	Is the ratio between the number of accepted demands and the total number of demands Eq. 1
Acceptance ratio per class AR_c	Is the ratio between the number of accepted demands by ($Class_c$) and the total number of demands by this ($Class_c$) Eq. 2
Blocking probability Bp	The ratio between the number of blocked demands (rejected) and the total number of demands
Blocking probability per class Bp_c	The ratio between the number of blocked demands by ($Class_c$) and the total number of demands by this ($Class_c$) Eq. 3
Total Utilization U	The ratio between the accepted resources and the total capacity of resources Eq. 5
Utilization per class U_c	The ratio between the accepted resources by ($Class_c$) and the total capacity of resources by this($Class_c$) Eq. 6

$$AR = AD/D \tag{1}$$

$$AR_c = AD_c/D_c \tag{2}$$

$$Bp = BD/D \tag{3}$$

$$Bp_c = BD_c/D_c \tag{4}$$

$$U = \frac{\sum_{j=1}^{D} d_j(CT_c)\, I_{A(j)}}{R} \tag{5}$$

Where $I_{A(j)}$ is an indicator function equal to 1 if j belongs to A and 0 otherwise. The set A(j) corresponds to total accepted demands.

$$U_c = \frac{\sum_{j=1}^{D_c} d_j(CT_c)\, I_{A_c(j)}}{RC_c} \tag{6}$$

Where $I_{A_c(j)}$ is an indicator function equal to 1 if j belongs to A_c and 0 otherwise. The set $A_c(j)$ corresponds to accepted demands by class c.

1) Example of Proposed Off-line SKM Algorithm: SKM was compared to RDM and AllocTC, in terms of user priorities and available resources in the system. In this example, the resources capacity of the system equal to 40 units and divided into four priority classes. Each class has the same amount of resources equal to 10 units. Nine demands to use available resources (i.e. 10, 10, 10 and 10) must be admitted into the system as follows:

#1: From S to D, 8 units priority 3
#2: From S to D, 4 units priority 3
#3: From S to D, 7 units priority 4
#4: From S to D, 7 units priority 4
#5: From S to D, 9 units priority 1
#6: From S to D, 6 units priority 2
#7: From S to D, 6 units priority 3
#8: From S to D, 7 units priority 2
#9: From S to D, 12 units priority 4

The overall performance of SKM in this example as shown in Table V demonstrates the performance of SKM in the offline case for the demands to be admitted on the given classes of the link. For example, the demand $\#9 : 12_4$ is admitted on the system where it used all resources from its priority class and borrowed two unused resources from class 3. Table VI shows the link load by TC, U_c, U, AR_c and AR results by using offline SKM. Which means, after the admission of the demands, we can calculate the link load for each class, the utilization of each class, and how many admitted or rejected demands in the system.

C. Online evaluation metrics

The metrics for the finite duration demands considered in our work, as defined in Table VII can be evaluated as follows:

TABLE VII: Online metrics definitions

Abbreviation	Explanation
Acceptance ratio AR(T)	The ratio between the number of accepted demands and the total number of demands until time T. Where the observation time (total consumed time by simulation) from t_0 until T Eq. 7
Acceptance ratio per class $AR_c(T)$	The ratio between the number of accepted demands by each class separately and the total number of demands by the same class until time T Eq. 8
Total Utilization: U(T)	The ratio between the accepted resources in all classes within a time duration T_j and the total capacity of resources at the time of observation Eq. 9
Utilization per class: $U_c(T)$	The ratio between the accepted resources by each class separately within T_j and the total capacity of resources of the same class at the time of observation Eq. 10

TABLE III: Technical behavior and operational characteristics comparison matrix

Behavioral characteristics	**MAM**	**RDM**	**AllocTC**	**SKM**
Efficient Resource utilisation with high traffic load of lower priority classes	Low	High	High	High
Efficient Resource utilisation with high traffic load of higher priority classes	Low	Low	High	Very High
Resource utilisation along the link	Low	Low (but better than MAM)	High	High
Accepted demands of higher priority classes along with the link	Low	Low	Low	Very High
Traffic classes isolation	High	Medium	Low	Low
Operational characteristics	**MAM**	**RDM**	**AllocTC**	**SKM**
P_{HTL}	No	Yes	Yes	Yes
P_{LTH}	No	No	Yes	No
K_i	No	No	No	Yes

TABLE V: SKM example (Off-line)

# of demand : d_p 4 priority classes	**Avialable Resources**	**SKM-Allocation**
$\#9 : 12_4$	(10,10,10,10)	(10,10,8,0) SL_3
$\#3 : 7_4$	(10,10,8,0)	(10,10,1,0) MAM
$\#4 : 7_4$	(10,10,1,0)	(10,4,0,0) SL_2
$\#1 : 8_3$	(10,4,0,0)	(6,0,0,0) SL_1
$\#7 : 6_3$	(6,0,0,0)	(0,0,0,0) SL_1
$\#2 : 4_3$	(0,0,0,0)	Rejected
$\#8 : 7_2$	(0,0,0,0)	Rejected
$\#6 : 6_2$	(0,0,0,0)	Rejected
$\#5 : 9_1$	(0,0,0,0)	Rejected
# of demand : d_p 4 priority classes	**Avialable Resources**	**AllocTC-Allocation**
$\#1 : 8_3$	(10,10,10,10)	(10,10,2,10)
$\#2 : 4_3$	(10,10,2,10)	(10,8,0,10)
$\#3 : 7_4$	(10,8,0,10)	(10,8,0,3)
$\#4 : 7_4$	(10,8,0,3)	(10,4,0,0)
$\#5 : 9_1$	(10,8,0,3)	(1,4,0,0)
$\#6 : 6_2$	(1,4,0,0)	(1,0,2,0) P_{LTH}, $\#2 : 4_3$ Rejected
$\#7 : 6_3$	(1,0,2,0)	Rejected
$\#8 : 7_2$	(1,0,2,0)	(0,0,0,3) P_{LTH}, $\#4 : 7_4$ Rejected
$\#9 : 12_4$	(0,0,0,3)	Rejected
# of demand : d_p 4 priority classes	**Avialable Resources**	**RDM-Allocation**
$\#1 : 8_3$	(10,10,10,10)	(2,10,10,10)
$\#2 : 4_3$	(2,10,10,10)	(0,8,10,10)
$\#3 : 7_4$	(0,8,10,10)	(0,1,10,10)
$\#4 : 7_4$	(0,1,10,10)	Rejected
$\#5 : 9_1$	(0,1,10,10)	(0,0,2,10)
$\#6 : 6_2$	(0,0,2,10)	(0,0,0,6)
$\#7 : 6_3$	(0,0,0,6)	Rejected
$\#8 : 7_2$	(0,0,0,6)	Rejected
$\#9 : 12_4$	(0,0,0,6)	Rejected

TABLE VI: SKM example (Off-line) Results

SKM Strategy	**Class 1**	**Class 2**	**Class 3**	**Class 4**	**Link**
Load by priority	10	10	10	10	40
Utilization (U)	U_1=0/10 =0%	U_2=0/10 =0	U_3=8+6/40 =35%	U_4=12+7+7/40 =65%	U=40/40 =100%
Blocking probability (Bp)	Bp_1=1/1	Bp_2=2/2	Bp_3=1/3	Bp_4=0/3	Bp=4/9
Acceptance ratio (AR)	AR_1=0/1	AR_2=0/2	AR_3=2/3	AR_4=3/3	AR=5/9
AllocTC Strategy	**Class 1**	**Class 2**	**Class 3**	**Class 4**	**Link**
Load by priority	10	10	10	7	37
Utilization (U)	U_1=9/40 =22.5%	U_2=6+7/40 =32.5%	U_3=8/40 =20%	U_4=7/40 =17.5%	U=37/40 =92.5%
Blocking probability (Bp)	Bp_1=0/1	Bp_2=0/2	Bp_3=2/3	Bp_4=2/3	Bp=4/9
Acceptance ratio (AR)	AR_1=1/1	AR_2=2/2	AR_3=1/3	AR_4=1/3	AR=5/9
RDM Strategy	**Class 1**	**Class 2**	**Class 3**	**Class 4**	**Link**
Load by priority	10	10	10	4	34
Utilization (U)	U_1=9/40 =22.5%	U_2=6/40 =15%	U_3=8+4/40 =30%	U_4=7/40 =17.5%	U=34/40 =85%
Blocking probability (Bp)	Bp_1=1/1	Bp_2=2/2	Bp_3=1/3	Bp_4=0/3	Bp=4/9
Acceptance ratio (AR)	AR_1=0/1	AR_2=0/2	AR_3=2/3	AR_4=3/3	AR=5/9

$$AR(T) = AD(T)/D(T) \tag{7}$$

$$AR_c(T) = AD_c(T)/D_c(T) \tag{8}$$

$$U(T) = \frac{\sum_{j=1}^{D} d_j(CT_c)\, I_{A(j)}\, T_j}{R * T} \tag{9}$$

$$U_c(T) = \frac{\sum_{j=1}^{D_c} d_j(CT_c)\, I_{A_c(j)}\, T_j}{RC_c * T} \tag{10}$$

Note that the definition of $I_{A(j)}$ and $I_{A_c(j)}$ for online scenario as in Eq. 5 and Eq. 6 respectively.

1) Online Simulation Scenarios: To evaluate our solution, the system used consists of a resource capacity equal to R = 160 units. Each class has RC_c=40 units. This resource capacity is divided into four classes considered in the system. The proposed strategy is used to check whether there are sufficient resources according to the class of the demand that needs to be admitted into the system, and then evaluate the metrics for comparison with other strategies for an online scenario. In the simulations, the demands are generated with a fixed lifetime equal 1-time slot, and the size is also fixed equal to 1 unit as the minimum granularity for allocation. Each demand has a single priority generated randomly from (1 to 4) with a generation rate of demands per each unit time equal to 200 demand. The total number of demands among classes generated until 100 unit time is 20,000 demands for each scenario. The traffic load consideration of the validation scenarios in each unit time is as follow: **Scenario 01:** Higher load in higher priority classes ($CT_1 = 20units > CT_2 = 40units > CT_3 = 60units > CT_4 = 80units$). **Scenario 02:** Higher load in all priority classes ($CT_4 = 50units > CT_3 = 50units > CT_2 = 50units > CT_1 = 50units$). Please also note that the used computer had Intel (R) Core (TM) 2 CPU 6400 @ 2.13GHz Memory 6GB and the used tool was Eclipse Java Oxygen.

V. OBTAINED RESULTS AND DISCUSSION

The performance of SKM is evaluated and compared with AllocTC and RDM in terms of the number of performance metrics as described below. The main objective of the above scenarios is to analyze the performance of SKM under different load distributions among the different priority classes.

TABLE VIII: Summary of scenario 1 results

Scenario1	Simulations results (Values in %)									
Metrics	U_1	U_2	U_3	U_4	U	AR_1	AR_2	AR_3	AR_4	AR
SKM	0	12.5	37.5	50	100	0	50	100	100	80
AllocTC	12.5	25	25	37.5	100	100	100	66.67	75	80
RDM	12.5	25	25	25	87.5	100	100	66.67	50	70

TABLE IX: Summary of scenario 2 results

Scenario2	Simulations results (Values in %)									
Metrics	U_1	U_2	U_3	U_4	U	AR_1	AR_2	AR_3	AR_4	AR
SKM	6.25	31.25	31.25	31.25	100	20	100	100	100	80
AllocTC	25	25	25	25	100	80	80	80	80	80
RDM	25	25	25	25	100	80	80	80	80	80

The obtained simulation results from scenario 1 are summarized in Table. VIII, in terms of AR_c, AR, U, U_c and shown in Fig. 3a for SKM, Fig. 3b for AllocTC and Fig. 3c for RDM. From the obtained results, the algorithms show a constant behavior in time since we assumed that 200 demands need to be allocated in each unit time along 100 unit times, on a single link with capacity equal to 160 resources (should cause link saturation). In light of that, the SKM outperforms RDM and AllocTC in the highest priority class by 50% and 25% in terms of AR_4, and by 25% and 12.5% in terms of U_4. AllocTC achieved higher acceptance ratio and utilization than RDM in class 4, since, in AllocTC performance, the higher priority classes can borrow unused resources from the lower ones to admit the demands (class 4 shared 20 resources from the lowest class). This is attributed to the fact that scenario one considered the higher priority classes to have more demand than the lower priority classes. Also, from the results, SKM outperforms RDM and AllocTC in class 3 by 33.33 % in terms of AR_3 and by 12.5% in terms of U_3 (as the expected from the behaviours). The SKM approach registers highest AR and U performance in the higher priority classes, due to the kicking operation as explained earlier. Moreover, even when the lower classes have fewer demands than the assigned resources for admitting demands, the unused resources can be shared by higher priority classes, which is not the case with RDM. If there are any unused resources in class 1 or 2 for the case of RDM, these resources will stay idle even if there is congestion in the higher priority classes.

In terms of total U and total AR, when we increase the load in higher priority classes, the RDM performance is the lowest one among the three strategies, achieving 70% as AR and 87.5% as U. Where the lower priority classes can only share resources from the higher ones. Therefore, in all unit times, the total acceptance ratio along the system will not exceed 160/200 = 80% as in SKM and AllocTC even if the number of demands was more than the capacity of the system. This is because each class cannot exceed its resources constraints (class 1 = 20 units, class 2 = 40 units, class 3 = 40 units, class 4 = 40 units).

Finally, from the results of scenario one, by increasing the number of demands in the higher priority classes we can realize a significant performance difference between SKM, AllocTC and RDM approach in terms of the strictness on priority. Thus, SKM provided better performance in terms of AR and U.

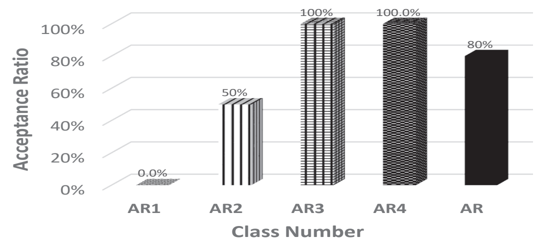
The obtained simulation results from scenario two are highlighted in Fig. 4 and summarized in Table IX. The results indicate that SKM, RDM and AllocTC, resulted in 100% U and 80% AR, where 160 demands are accepted from 200 demands per each unit time. From the obtained results in this scenario, the algorithms also show constant behavior in time. As expected, SKM registered the highest performance among the other two strategies (RDM, AllocTC) by 20% in terms of AR_4. Similarly, SKM outperforms RDM and AllocTC by 20% in terms of AR_3. Further, in terms of U_c, SKM, achieved 6.5% for class 4 and, 6.5% for class 3 more than both RDM and AllocTC. The above results show a superior performance of SKM for class 4 and 3 in terms of both AR_c and U_c. This can be justified by the nature of SKM, which permits higher priority classes to share unused resources from the lower ones and vice versa. The results also reveal that RDM has the same performance as AllocTC for the above classes under the considered scenario in terms of both AR_c and U_c. This can also be justified by the nature of AllocTC, which permits lower priority classes to share unused resources from the higher ones and vice versa similar to our proposal. However, in case of system saturation, unlike SKM, all borrowed resources should be returned in both senses for AllocTC case. Therefore, as illustrated in this scenario settings with the same traffic load in all classes, each class accepted 40 demands from 60 demands that needed to be admitted. In terms of RDM performance, the higher priority classes can not share unused resources from the lower ones, so it had the same equivalent performance to AllocTC.

SKM achieves the lowest performance in lower classes due to the kicking operation, which results in expelling the lower priority users to satisfy the demand requirements of the high priority classes. On the other hand, SKM intends to favour users belonging to high priority classes in terms of admission and resource allocation, hence the observed superior performance for high classes at the expense of low priority classes. Moreover, this behaviour makes SKM a right candidate for prioritized admission control.

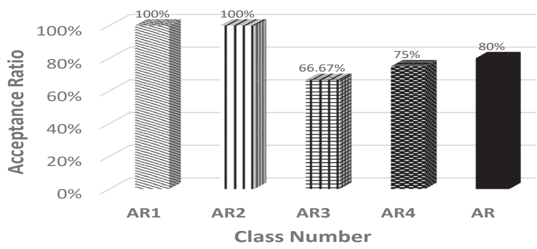
From the considered scenarios, SKM can guarantee to achieve 100% AR_c as long as the demanded resources from higher priority classes not exceed the capacity of the system. It also registers a better overall resource utilization compared to RDM in both traffic scenarios and the same performance as AllocTC. These results justify that SKM is a better admission control model for prioritized services than the existing schemes based in BAMs.

VI. CONCLUSIONS AND FUTURE WORK

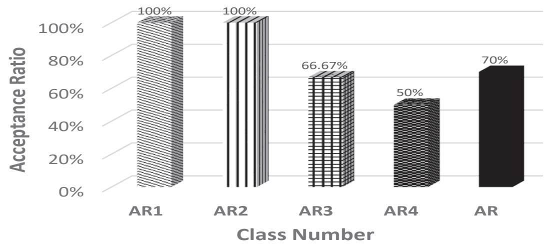
In this paper, a novel admission control model has been proposed, able to guarantee 100% utilization under different



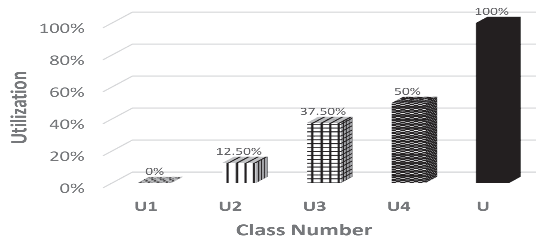
(a) SKM Acceptance Ratio per Class



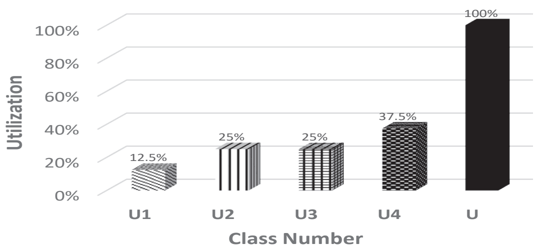
(b) AllocTC Acceptance Ratio per Class



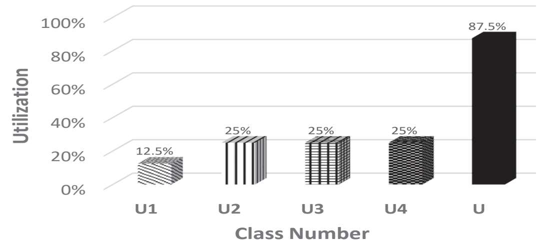
(c) RDM Acceptance Ratio per Class



(d) SKM Utilization per Class

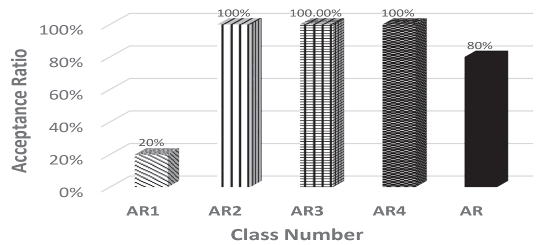


(e) AllocTC Utilization per Class

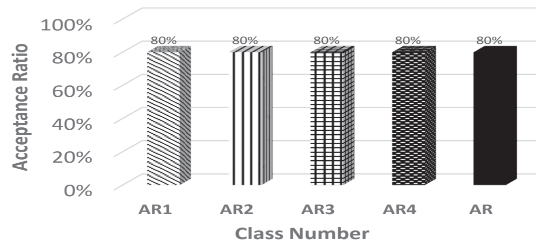


(f) RDM Utilization per Class

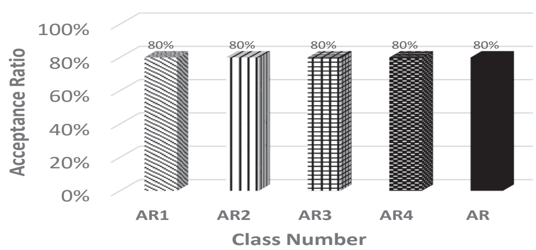
Fig. 3: SKM, AllocTC and RDM AR_c, U_c Comparison of Scenario 01



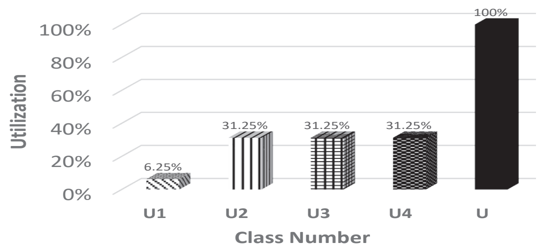
(a) SKM Acceptance Ratio per Class



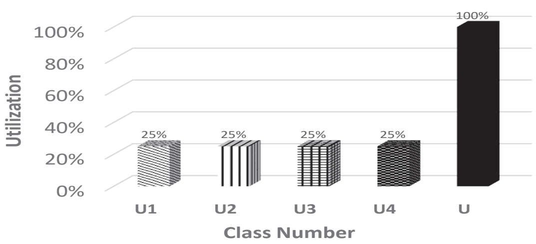
(b) AllocTC Acceptance Ratio per Class



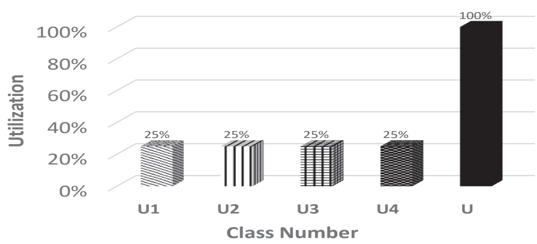
(c) RDM Acceptance Ratio per Class



(d) SKM Utilization per Class



(e) AllocTC Utilization per Class



(f) RDM Utilization per Class

Fig. 4: SKM, AllocTC and RDM AR_c, U_c Comparison of Scenario 02

priorities consideration, specially designed for highly congested scenarios with strict constraints for priority classes.

On the other hand, for the case of uncongested scenarios the SKM behaves similar to MAM, RDM and AllocTC. In RDM, the reservation of resources is made from bottom to top and not the reverse. So, in this way, resources utilization is more effective in comparison to MAM, which does not permit resource sharing across classes, but there is no guaranteed bandwidth for higher priority classes. Therefore, the benefit of using SKM is that the given class can be accepted regarding other classes (high or low) by means of initiating a squatting process, this is similar to the AllocTC per link behaviour of traffic distribution scenario. Beyond that, in SKM, the usage of resources for the higher priority classes is greater than originally reserved. SKM guarantees 100 percent of admission of high priority demands as long as there are resources in the lower priority classes, regardless of whether these resources are unused or occupied by the lower priority classes by means of initiating a kicking process. It is expected that groups of higher priority applications on multi-service networks could benefit from improved link utilization achieved by SKM. This corresponds to dynamically providing support to improve the quality of the application (SLA) for traffic distributions that occur in actual system operation, which means that the SKM is strict on priorities more than AllocTC and RDM.

Simulations validated the performance in the considered system in terms of utilization and acceptance ratio, including metrics per priority class, such as in scenario one SKM outperforms RDM and AllocTC in the highest priority class by 50% and 25%, in terms of AR_4, and by 25% and 12.5% respectively in terms of U_4. Also, SKM outperforms RDM and AllocTC in class 3 by 33.33 %, in terms of AR_3 and by 12.5% in terms of U_3. In terms of total U and total AR, when we increase the load in higher priority classes, the RDM performance is the lowest one among the three strategies, achieving 70% as AR and 87.5% as U compared to 80% AR and 100% U in both AllocTC and SKM.

As future work, the authors are planning to extend the SKM to consider other scenarios to study more the behaviour of SKM, as well as studying the complexity of the SKM implementation and propose a fast heuristic of SKM. As another future work, SKM will be improved by considering aforementioned thresholds to define and guarantee minimum resources for each class that will avoid resources beat down for lower priority classes.

ACKNOWLEDGMENT

This work has been partially supported by the Ministerio de Economy Competitividad of the Spanish Government under project TEC2016-76795-C6-1-R and AEI/FEDER, UE, and the SGR project, grant number 2017 SGR 397, from the Generalitat de Catalunya.

REFERENCES

[1] H. Zhang et al., “Network Slicing Based 5G and Future Mobile Networks: Mobility, Resource Management, and Challenges,” in IEEE Commun. Mag., vol. 55, no. 8, pp. 138-145, Aug. 2017.

[2] P. Caballero, A. Banchs, G. de Veciana, X. Costa-Pérez and A. Azcorra, “Network Slicing for Guaranteed Rate Services: Admission Control and Resource Allocation Games,” in IEEE Transactions on Wireless Communications, vol. 17, no. 10, pp. 6419-6432, Oct. 2018.

[3] B. Han et al., “Admission and Congestion Control for 5G Network Slicing,” 2018 IEEE Conference on Standards for Communications and Networking (CSCN), Paris, 2018, pp. 1-6.

[4] D. T. Hoang, D. Niyato, P. Wang, A. De Domenico and E. C. Strinati, “Optimal Cross Slice Orchestration for 5G Mobile Services,” 2018 IEEE 88th Vehicular Technology Conference (VTC-Fall), Chicago, IL, USA, 2018, pp. 1-5.

[5] M. Jiang, M. Condoluci and T. Mahmoodi, “Network slicing management and prioritization in 5G mobile systems,” European Wireless 2016; 22th European Wireless Conference, Oulu, Finland, 2016, pp. 1-6.

[6] R.F. Reale, R.Bezerra, J.Martins, “A preliminary evaluation of bandwidth allocation model dynamic switching,” Int. J. Comput. Netw. Commun. (IJCNC), vol. 6, no. 3, pp. 131-143, May. 2014.

[7] G. M. Duraes et al., “Evaluating the applicability of bandwidth allocation models for EON slot allocation,” in IEEE Intl. Conf. on Adv. Net. and Tel. Sys. (ANTS), Bhubaneswar, pp. 1-6, 2017.

[8] F. Le Faucheur, W. Lai, “Maximum Allocation Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering,” RFC 4125, 2005.

[9] F. Le Faucheur, “Russian Dolls Bandwidth Constraints Model for Diffserv-aware MPLS Traffic Engineering,” RFC 4127, 2005.

[10] D. Adami, C. Callegari, S. Giordano, M. Pagano, M. Toninelli, “G-RDM: a new bandwidth constraints model for DS-TE networks,” in: Proceedings of the IEEE Global Telecommunications Conference, 2007, pp. 2472-2476.

[11] R.F. Reale, W. d. C. P. Neto and J. S. B. Martins, “AllocTC-sharing: A new bandwidth allocation model for DS-TE networks,” in: Proc. of the IEEE Net. Oper. and Mgmt. Symposium, pp. 1-4, Quito, 2011.

[12] R.F. Reale, Rafael Freitas, Romildo Martins da Silva Bezerra and Joberto S. B. Martins, “G-BAM: A Generalized Bandwidth Allocation Model for IP/MPLS/DS-TE Networks,” CoRR abs/1806.07292 (2014): n. pag.

[13] S.K. Sadon, N.M. Din, M.H. Al-Mansoori, N.A. Radzi, I.S. Mustafa, M. Yaacob, M.S.A. Majid, “Dynamic hierarchical bandwidth allocation using Russian Doll Model in EPON,” Comput. Electr. Eng. 38 (6) (2012) 1480-1489.

[14] R. Trivisonno, R. Guerzoni, I. Vaishnavi, A. Frimpong, “Network resource management and QoS in SDN-enabled 5G systems," in: Proceedings of the IEEE Global Communications Conference, 2015, pp. 1–7.

[15] N. Subhashini, A.B. Therese, “User prioritized constraint free dynamic bandwidth allocation algorithm for EPON networks,” Indian J. Sci. Technol. 8 (33) (2015) 1–7.

[16] A. Ayoub Bahnasse et al., “Novel SDN architecture for smart MPLS Traffic Engineering-DiffServ Aware management,” in Future Generation Computer Systems, Volume 87, pp. 115-126, 2018.

[17] X. Hesselbach et al., “Management of resources under priorities in EON using a modified RDM based on the squatting-kicking approach,” Intl. Conf. on Transparent Optical Networks (ICTON), Trento, pp. 1-5, 2016.